

MUSIC DISPLAY CONTROLLER AND METHOD, AND MEMORY MEDIUM

Publication number: JP2002258837

Publication date: 2002-09-11

Inventor: HIRATSUKA MASARU

Applicant: YAMAHA CORP

Classification:

- International: G10G1/00; G09B15/02; G09B15/04; G10H1/00;
G10G1/00; G09B15/02; G10H1/00; (IPC1-7): G10G1/00

- European: G09B15/02C; G09B15/04

Application number: JP20010054923 20010228

Priority number(s): JP20010054923 20010228

Also published as:



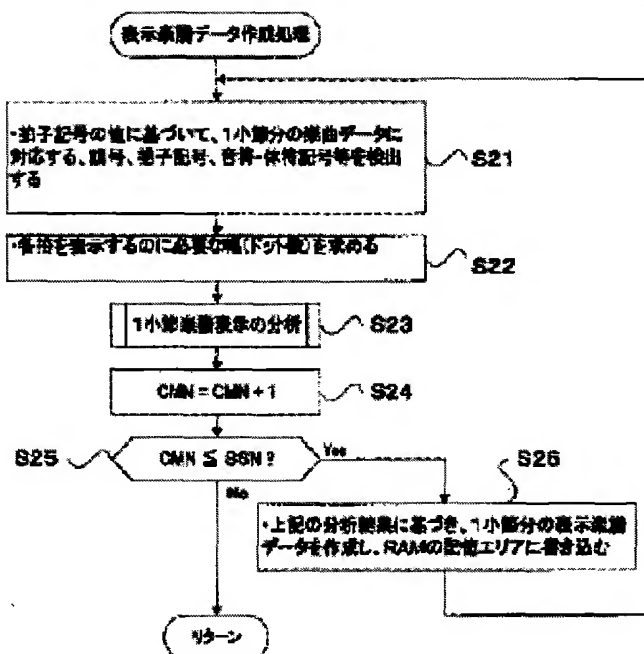
US6545208 (B2)

US2002118562 (A1)

Report a data error here

Abstract of JP2002258837

PROBLEM TO BE SOLVED: To make music display of high visibility with respect to a user.
SOLUTION: Display means is for displaying the music score for playing music. Analyzing means analyzes the music data relating to the playing music supplied from supplying means and detects the music score signs included in prescribed ranges by each of the prescribed ranges of the playing music. Control means arranges the music score signs included within one prescribed range within one or plural basic display blocks, thereby displaying the music score signs included within the respective prescribed ranges by each of the prescribed range. The music score signs can then be displayed on the display means, without superposing the music score signs on each other or without making the music score signs come to proximity of each other even in case of the occurrence of such an incident that the music score signs included within the one prescribed ranges cannot be arranged fully within one basic display block, so that the user is able to obtain the music score having always the high visibility.



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2002-258837
(P2002-258837A)

(43)公開日 平成14年9月11日(2002.9.11)

(51)Int.Cl. ⁷	識別記号	F I	テマコード [*] (参考)
G 1 0 G 1/00		G 1 0 G 1/00	5 D 0 8 2
G 1 0 H 1/00	1 0 2	G 1 0 H 1/00	1 0 2 Z 5 D 3 7 8

審査請求 未請求 請求項の数11 O L (全 16 頁)

⇒ (21)出願番号 特願2001-54923(P2001-54923)

(22)出願日 平成13年2月28日(2001.2.28)

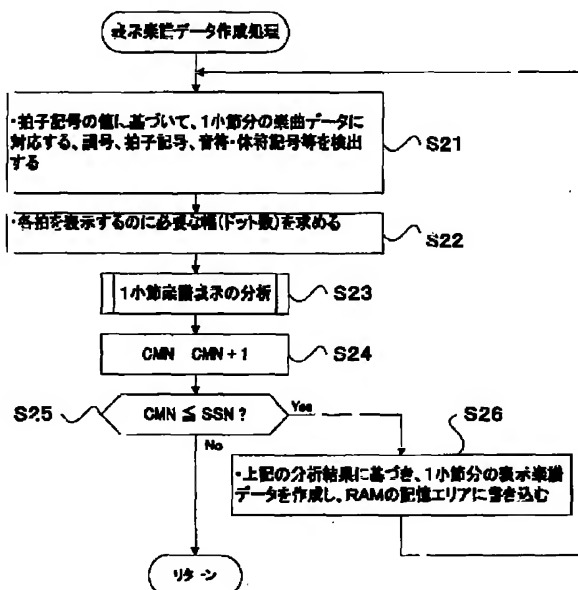
(71)出願人 000004075
ヤマハ株式会社
静岡県浜松市中沢町10番1号
(72)発明者 平塚 賢
静岡県浜松市中沢町10番1号 ヤマハ株式
会社内
(74)代理人 10007/539
弁理士 飯塚 義仁
Fターム(参考) 5D082 AA03 AA04 AA05 AA06 AA07
AA26 AA28
5D378 MM09 MM12 NN02 NN25 TT02
TT03 TT23

(54)【発明の名称】 楽譜表示制御装置及び方法並びに記憶媒体

(57)【要約】

【課題】 ユーザに対して視認性のよい楽譜表示を行う。

【解決手段】 表示手段は、演奏曲の楽譜を表示する。分析手段は、供給手段から供給された演奏曲に関する楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する。制御手段は1つの所定範囲内に含まれる楽譜記号を1又は複数の基本表示ブロック内に配置することで、各所定範囲内に含まれる楽譜記号を所定範囲毎に可変する表示幅に表示する。こうすると、1つの所定範囲内に含まれる楽譜記号を1つの基本表示ブロック内に配置しきれないようなことが生じた場合であっても、楽譜記号を重複あるいは近接させることなく表示手段に表示することができることから、ユーザは常に視認性のよい楽譜を得ることができる。



【特許請求の範囲】

【請求項1】 表示手段と、
演奏曲に関する楽曲データを供給する供給手段と、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する分析手段と、
前記表示手段において所定の基本表示ブロックを単位として楽譜を表示させる制御を行うものであって、1つの前記所定範囲内に含まれる楽譜記号を1又は複数の前記各基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示する制御手段とを具えた楽譜表示制御装置。

【請求項2】 前記制御手段は、1小節内に含まれる楽譜記号を表示するために必要な表示幅と前記基本表示ブロックの表示幅との比較結果に応じて1小節内に含まれる楽譜記号を拍毎に分割し、該分割した楽譜記号を各々の基本表示ブロックに配置することを特徴とする請求項1に記載の楽譜表示制御装置。

【請求項3】 前記制御手段は、隣り合う基本表示ブロックに各々表示する楽譜記号の表示密度がより均等になるように楽譜記号を配置することを特徴とする請求項1又は2に記載の楽譜表示制御装置。

【請求項4】 前記制御手段は、1つの前記所定範囲内に含まれる楽譜記号のうちの一部の楽譜記号を異なる段の基本表示ブロックに配置する場合には、該異なる段の基本表示ブロックの始めに所定の楽譜記号を追加することを特徴とする請求項1乃至3のいずれかに記載の楽譜表示制御装置。

【請求項5】 表示手段と、
演奏曲に関する楽曲データを供給する供給手段と、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する分析手段と、
前記表示手段において所定の基本表示ブロックを単位として楽譜を表示させる制御を行うものであって、1つの前記所定範囲内に含まれる楽譜記号を1又は複数の前記各基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示する制御手段と、
演奏曲の演奏進行にあわせて演奏タイミングを前記楽譜記号の表示に対応するように表示する演奏タイミング表示制御手段とを具えた楽譜表示制御装置。

【請求項6】 前記演奏タイミング表示制御手段は、演奏タイミングの表示移動速さを基本表示ブロックに配置された楽譜記号の数に応じて変化させることを特徴とする請求項5に記載の楽譜表示制御装置。

【請求項7】 前記制御手段は、検出した楽譜記号に所定の拍子記号が含まれる場合に該拍子記号を基本表示ブロックに配置しないことにより、前記表示手段に該拍子記号を表示しないことを特徴とする請求項1乃至6のい

ずれかに記載の楽譜表示制御装置。

【請求項8】 演奏曲に関する楽曲データを供給するステップと、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出するステップと、
1つの前記所定範囲内に含まれる楽譜記号を1又は複数の所定の基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示するステップとを具える楽譜表示制御方法。

【請求項9】 演奏曲に関する楽曲データを供給するステップと、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出するステップと、
1つの前記所定範囲内に含まれる楽譜記号を1又は複数の所定の基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示するステップと、
演奏曲の演奏進行にあわせて演奏タイミングを前記楽譜記号の表示に対応するように表示するステップとを具える楽譜表示制御方法。

【請求項10】 機械読取り可能な記憶媒体であって、演奏曲の楽譜を表示する方法をプロセッサに実行させるためのプログラムを記憶してなり、前記楽譜表示制御方法は、
演奏曲に関する楽曲データを供給するステップと、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出するステップと、
1つの前記所定範囲内に含まれる楽譜記号を1又は複数の所定の基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示するステップとを具備する。

【請求項11】 機械読取り可能な記憶媒体であって、演奏曲の楽譜を表示する方法をプロセッサに実行させるためのプログラムを記憶してなり、前記楽譜表示制御方法は、
演奏曲に関する楽曲データを供給するステップと、
前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出するステップと、
1つの前記所定範囲内に含まれる楽譜記号を1又は複数の所定の基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示するステップと、
演奏曲の演奏進行にあわせて演奏タイミングを前記楽譜記号の表示に対応するように表示するステップとを具備する。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、所定の楽曲データに基づいて演奏される楽曲の進行にあわせて該楽曲の楽譜を表示する楽譜表示制御装置及び方法並びに記憶媒体に関し、特に高度な曲構成をした楽曲の楽譜を表示する場合にユーザに対して最も視認性のよい楽譜表示を行う楽譜表示制御装置及び方法に関する。

【0002】

【従来の技術】従来から知られているように、電子楽器やパーソナルコンピュータ等を用いてユーザ所望の楽曲データに基づいて自動演奏（つまり、楽曲データの再生）を行う際には、所定の画面上に当該演奏曲の楽譜を自動的に表示することが行われている。すなわち、自動演奏される演奏曲にあわせて同時にピアノやギター等の楽器演奏を楽しみたい、あるいは楽器演奏の練習をしたいといったようなユーザのために、現在演奏進行中の曲部分に対応する楽譜を所定区間ずつ所定の表示態様で（例えば、4小節分の楽譜を1段に表示するものを1画面につき2段ずつ表示するなど）演奏曲の進行にあわせて順次に表示することが行われている。このようにすると、ユーザ自身がわざわざ曲毎に楽譜を用意しなくてもよいことから便利である。さらに、デモ曲などのオリジナルのアレンジなどがなされた場合には対応する楽譜がそもそも存在しないことから、こうした場合にも曲に対応する楽譜を表示することから非常に便利である。また、所定の表示形態のカーソルを演奏曲の進行にあわせて楽譜上を移動するように表示することで、ユーザに対してそのカーソルが移動した先の楽譜にあわせてピアノやギター等の楽器を操作するように指示することも、演奏曲の楽譜表示に加えて行われている。以上のようにすることで、ユーザはすぐにも演奏曲にあわせて楽器演奏などの練習を一人で始めることができるようになっていく。こうした楽譜やカーソル等の表示制御は、楽譜表示制御装置を用いることによって行われる。

【0003】

【発明が解決しようとする課題】ところで、従来の楽譜表示制御装置によって行われている楽譜表示は、子供や初心者向けの簡単な曲構成である楽曲の楽譜を表示することを想定した楽譜表示である場合が多いことから、所定の画面上において固定的な表示幅を持つ基本表示ブロック内に1小節分の楽曲に対応する楽譜の全てを表示するものであった（これにより、各小節を区切るための小節線は各小節毎に予め固定された所定の表示位置に表示される）。そのために、1小節内に音部記号や拍子記号や調号あるいは音符や休符といった楽譜記号が数多く存在し、該1小節内の全ての楽譜記号を表示するのに必要な表示幅が基本表示ブロックの持つ表示幅よりも大きくなってしまふような場合（例えば、音楽レベルの高いユーザ向けである高度な曲構成である楽曲の楽譜を表示す

るような場合）には、該基本表示ブロック内に表示しきれない楽譜記号が生じてしまうことがある、という問題点があった。このような問題点を解決する1つの方法として、高度な曲構成である楽曲の楽譜を表示する場合には、表示が前後する楽譜記号を無理やりに重ねて表示することによって、該基本表示ブロック内に1小節分の全ての楽譜記号を表示しようとする方法が従来から知られている。しかし、このように一部の楽譜記号を重ねるようにして楽譜表示を行う方法によると、ユーザにとって非常に見にくい楽譜が表示されることになる。すなわち、ユーザの楽譜の視認性という点において非常に不都合な楽譜表示が行われる、という問題点があった。

【0004】本発明は上述の点に鑑みてなされたもので、演奏曲の進行にあわせて該曲の楽譜を適宜に見やすい表示態様で所定の画面上に順次に表示していくことによって、1小節中に著しく多数の楽譜記号が存在する高度な曲構成をした楽曲であっても、ユーザが演奏の進行にあわせて確実に楽譜記号を視認することのできる楽譜表示制御装置及び方法を提供しようとするものである。

【0005】

【課題を解決するための手段】本発明に係る楽譜表示制御装置は、表示手段と、演奏曲に関する楽曲データを供給する供給手段と、前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する分析手段と、前記表示手段において所定の基本表示ブロックを単位として楽譜を表示させる制御を行うものであって、1つの前記所定範囲内に含まれる楽譜記号を1又は複数の前記各基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変する表示幅に表示する制御手段とを具備する。

【0006】表示手段は、演奏曲の楽譜を表示することのできるものである。制御手段は、前記表示手段において所定の基本表示ブロックを単位として楽譜を表示させる制御を行うものである。すなわち、前記基本表示ブロックは演奏曲を構成する所定の楽譜記号を複数表示することが可能となっている。分析手段は、供給手段から供給された演奏曲に関する楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する。こうして検出された前記所定範囲内に含まれる楽譜記号を制御手段によって前記各基本表示ブロック内に配置することにより、前記表示手段に演奏曲の楽譜を表示することができる。制御手段は1つの前記所定範囲内に含まれる楽譜記号を1又は複数の前記各基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示する。すなわち、1つの前記所定範囲内に含まれる楽譜記号を1つの基本表示ブロック内に配置しきれないようなことが生じた場合には、検出した楽譜記号を1又は複数の前記各基本表示ブロック内に配置する。これにより、楽譜記号

を重複表示あるいは近接表示しなければならなかった演奏曲の楽譜を、重複表示あるいは近接表示することなく楽譜を表示することができることから、ユーザにとって視認性のよい楽譜表示を表示手段上で実現することができるようになる。

【0007】また、本発明に係る楽譜表示制御装置は、表示手段と、演奏曲に関する楽曲データを供給する供給手段と、前記供給された楽曲データを分析し、演奏曲の所定範囲毎に該範囲内に含まれる楽譜記号を検出する分析手段と、前記表示手段において所定の基本表示ブロックを単位として楽譜を表示させる制御を行うものであって、1つの前記所定範囲内に含まれる楽譜記号を1又は複数の前記各基本表示ブロック内に配置することで、前記各所定範囲内に含まれる楽譜記号を所定範囲毎に可変の表示幅を用いて表示する制御手段と、演奏曲の演奏進行にあわせて演奏タイミングを前記楽譜記号の表示に対応するように表示する演奏タイミング表示制御手段とを具備する。これによると、楽譜表示と共に該楽譜表示の表示形態に応じて演奏タイミングを表示することから、ユーザに対して適切な演奏補助を行える点で有利である。

【0008】本発明は、装置の発明として構成し、実施することができるのみならず、方法の発明として構成し、実施することができる。また、本発明は、コンピュータまたはDSP等のプロセッサのプログラムの形態で実施することができるし、そのようなプログラムを記憶した記憶媒体の形態で実施することもできる。

【0009】

【発明の実施の形態】以下、この発明の実施の形態を添付図面に従って詳細に説明する。

【0010】図1は、この発明に係る楽譜表示制御装置を内蔵した電子楽器の実施の形態を示すハード構成ブロック図である。この実施形態に示す電子楽器は、マイクロプロセッサユニット(CPU)1、リードオンリメモリ(ROM)2、ランダムアクセスメモリ(RAM)3からなるマイクロコンピュータの制御の下に各種の処理が実行されるようになっていいる。この実施の形態では、1つのCPU1によって楽曲データに基づいて楽曲を自動的に演奏する自動演奏処理や自動演奏される楽曲の楽譜を演奏の進行にあわせて表示する楽譜表示処理等の各種処理を行う電子楽器を例に説明する。CPU1は、この電子楽器全体の動作を制御するものである。このCPU1に対して、データ及びアドレスバス1Dを介してリードオンリメモリ(ROM)2、ランダムアクセスメモリ(RAM)3、点灯制御回路4、検出回路5、スイッチ検出回路6、表示回路7、音源回路8、外部記憶装置9、MIDIインタフェース(I/F)10、通信インタフェース(I/F)11がそれぞれ接続されている。更に、CPU1には、タイマ割込み処理(インタラプト処理)における割込み時間や各種時間を計時するタイマ

1Aが接続されている。すなわち、タイマ1Aは時間間隔を計数したり、自動演奏のテンポを設定したりするためのテンポクロックパルスを発生するものである。タイマ1AからのテンポクロックパルスはCPU1に対してインタラプト命令として与えられ、CPU1はタイマ割込み処理(インタラプト処理)により自動演奏処理を実行する。

【0011】ROM2は、CPU1により実行あるいは参照される各種プログラムや各種データ等を格納するものである。RAM3は、自動演奏に関する各種演奏情報やCPU1がプログラムを実行する際に発生する各種パラメータなどを一時的に記憶するワーキングメモリとして、あるいは現在実行中のプログラムやそれに関連するデータを記憶するメモリ等として使用される。RAM3の所定のアドレス領域がそれぞれの機能に割り当てられ、レジスタやフラグ、テーブル、メモリなどとして利用される。発光ダイオード(LED)4Aは後述の演奏操作子5Aに対応するようにして配置されており、自動演奏実行の際に楽曲の進行に従って点灯・消灯することによって、ユーザに対してユーザが次に演奏操作すべき演奏操作子5Aを指示するためのものである。点灯制御回路4は、CPU1の制御のもと、演奏操作子5Aに対応するようにして配置された発光ダイオード(LED)4A毎に、連続的またはパルス的に電圧を印加して所定の発光ダイオード(LED)4Aを点灯・消灯させることができ、かつ、印加する電圧の大きさを調節してその発光ダイオード(LED)4Aの輝度を任意に設定することが可能な回路である。つまり、この実施例に示した電子楽器では、ユーザに対する演奏補助を発光ダイオード(LED)4Aの点灯・消灯によって行う。

【0012】演奏操作子5Aは楽音の音高を選択するための複数の鍵を備えた、例えば鍵盤等のようなものであり、各鍵に対応してキースイッチを有しており、この演奏操作子5A(鍵盤等)は楽音演奏のために使用できるのは勿論のこと、自動演奏を行う際に用いるメロディのピッチやリズムを入力するための入力手段として使用することもできる。検出回路5は、演奏操作子5Aの各鍵の押圧及び離鍵を検出することによって検出出力を生じる。パネル操作子(スイッチ等)6Aは自動演奏の対象となる楽曲データの選択、楽譜表示モードの選択、演奏ガイドモードの選択、自動演奏モードの選択などを行ったり、あるいは楽曲データに基づく自動演奏を行う際に用いる各種演奏条件等を入力するための各種のスイッチ等を含んで構成される。勿論、演奏操作子5Aは、音高、音色、効果等を選択・設定・制御するために用いる数値データ入力用のテンキーや文字データ入力用のキーボードなどの各種操作子を含んでよい。スイッチ検出回路6は、パネル操作子6Aの各スイッチの操作状態を検出し、その操作状態に応じたスイッチ情報をデータ及びアドレスバス1Dを介してCPU1に出力する。表

示回路7は自動演奏中の楽曲に関する楽譜等の各種情報を、例えば液晶表示パネル(LCD)やCRT等から構成されるディスプレイ7Aに表示するのは勿論のこと、自動演奏を行う際における各種演奏条件あるいはCPU1の制御状態などをディスプレイ7Aに表示する。この実施例に示す電子楽器においては、ディスプレイ7A上に演奏曲の進行に応じた楽譜が適宜に表示される。

【0013】音源回路8は、複数のチャンネルで楽音信号の同時発生が可能であり、データ及びアドレスバス1Dを経由して与えられた楽曲データを入力し、このデータに基づいて楽音信号を発生する。音源回路8から発生された楽音信号は、アンプやスピーカなどを含むサウンドシステム8Aから発音される。音源回路8から発生された楽音信号に対して、効果回路など(図示せず)を用いて所定の効果を付与するようにしてもよい。楽曲データの形式はMIDI形式のようなデジタル符号化されたものであってもよいし、PCM、DPCM、ADPCMのような波形サンプルデータ方式からなるものであってもよい。この音源回路8と効果回路(図示せず)とサウンドシステム8Aの構成には、従来のいかなる構成を用いてもよい。

【0014】外部記憶装置9は、楽曲データなどの制御情報やCPU1が実行する各種プログラム等の制御に関するデータ等を記憶するものである。前記ROM2に制御プログラムが記憶されていない場合、この外部記憶装置9(例えばハードディスク)に制御プログラムを記憶させておき、それを前記RAM3に読み込むことにより、ROM2に制御プログラムを記憶している場合と同様の動作をCPU1にさせることができる。このようにすると、制御プログラムの追加やバージョンアップ等が容易に行える。なお、外部記憶装置9はハードディスク(HD)に限られず、フロッピーディスク(FD)、コンパクトディスク(CD-ROM・CD-RAM)、光磁気ディスク(MO)、あるいはDVD(Digital Versatile Diskの略)等の着脱自在な様々な形態の外部記憶媒体を利用する記憶装置であってもよい。あるいは、半導体メモリなどであってもよい。

【0015】MIDIインタフェース(I/F)10は、他のMIDI機器10A等からMIDI規格の楽曲データ(MIDIデータ)などを当該電子楽器へ入力したり、あるいは当該電子楽器からMIDI規格の楽曲データ(MIDIデータ)などを他のMIDI機器10A等へ出力するためのインタフェースである。通信インタフェース11は、例えばLANやインターネット、電話回線等の通信ネットワークXに接続されており、概通信ネットワークXを介して、サーバコンピュータ11Aと接続され、当該サーバコンピュータ11Aから制御プログラムや楽曲データなどの各種データを電子楽器側に取り込むためのインタフェースである。すなわち、ROM

2や外部記憶装置9(例えばハードディスクなど)に制御プログラムや各種データが記憶されていない場合に、サーバコンピュータ11Aから制御プログラムや各種データをダウンロードするために用いられる。クライアントとなる電子楽器は、通信インタフェース11及び通信ネットワークXを介してサーバコンピュータ11Aへと制御プログラムや各種データのダウンロードを要求するコマンドを送信する。サーバコンピュータ11Aは、このコマンドを受け、要求された制御プログラムや各種データを、通信ネットワークXを介して本電子楽器へと配信し、本電子楽器が通信インタフェース11を介して、これら制御プログラムや各種データを受信して外部記憶装置9(例えばハードディスクなど)に蓄積することにより、ダウンロードが完了する。

【0016】なお、MIDIインタフェース10は専用のMIDIインタフェースを用いるものに限らず、RS-232C、USB(ユニバーサル・シリアル・バス)、IEEE1394(アイトリプルイー1394)等の汎用のインタフェースを用いてMIDIインタフェース10を構成するようにしてもよい。この場合、MIDIイベントデータ以外のデータをも同時に送受信するようにしてもよい。MIDIインタフェース10として上記したような汎用のインタフェースを用いる場合には、他のMIDI機器10AはMIDIイベントデータ以外のデータも送受信できるようにしてよい。勿論、音楽情報に関するデータフォーマットはMIDI形式のデータに限らず、他の形式であってもよく、その場合はMIDIインタフェース10と他のMIDI機器10Aはそれにあった構成とする。

【0017】ここで、図1に示した電子楽器において自動演奏を実行する際に用いられる楽曲データの構造について、図2を用いて説明する。図2は、図1に示した電子楽器で用いられる楽曲データのデータ構造の一実施例を示した概略図である。ただし、この実施例においては、楽曲データ構造として複数の異なる曲の楽曲データ1~楽曲データnを連続的に記憶したものを図示した。

【0018】個々の楽曲データ(楽曲データ1~n)は自動演奏や楽譜表示の際に用いられる楽音発生あるいは楽譜表示のための制御情報として用いられるデータであって、該楽曲データは内容別に初期設定のためのイベントデータとノートイベントデータ(ノートイベント1...)とデュレーションデータ(デュレーション1...)及びエンドコードとに分けることができる。初期設定のためのイベントデータは、音色やテンポ、あるいは調号や拍子などの該楽曲データを再生する際に用いられる各種音楽情報に関するイベントデータである。楽譜表示の際における調号の表示は、調号が初期設定のためのイベントデータに定義されている場合には該データに従って調号を表示する。調号が初期設定のためのイベントデータに定義されていない場合には、楽曲データ全体から調号

を自動的に検出し、該検出した調号を表示する。あるいは、予め定義された所定の調（例えば、「ハ長調」など）に従って調号を表示する。楽譜表示の際における拍子記号の表示も上記調号の場合と同様であって、初期設定のためのイベントデータに拍子が定義されている場合には該データに従って拍子記号を表示するし、初期設定のためのイベントデータに拍子が定義されていない場合には楽曲データ全体から自動的に検出した拍子に従う拍子記号を表示する。

【0019】ノートイベントデータ（ノートイベント1…）は、ノートオンやノートオフなどのノート（つまり音符）に関するイベントを表す演奏イベントデータである。該ノートイベントデータは、ノート番号とゲートタイムとベロシティなどのデータから構成される。ノート番号は音程を表すために半音単位に便宜的に付けられている番号（例えば、C2に対して「48」、C3に対して「60」など）であり、これにより楽譜表示の際には適切な音高位置を示す表示位置に音符記号が配置される。ゲートタイムは該ノートの発音時間長さを表すデータであり、これにより楽譜表示の際に適切な音符種類の音符記号（例えば、4分音符や16分音符など）が前記音高位置に配置されることになる。ベロシティは、音の強弱を表すデータである。デュレーションデータ（デュレーション1…）は、ノートイベントデータの発生タイミングを曲の先頭や各小節の先頭からの絶対時間、あるいは1つ前のノートイベントデータからの相対時間で示すためのデータである。これにより楽譜表示の際には、曲の先頭や小節線からの絶対的な表示位置、あるいは隣り合う音符記号との間における相対的な表示位置が決定され、該表示位置に基づいて音符記号が配置される。また、このような場合にゲートタイムとの関係から音符記号のほかに自動的に休符記号を追加して配置するようにしてもよい。勿論、休符に関するノートイベントデータを予め楽曲データ内に記憶してある場合には、こうした休符記号の自動的な追加は行わなくてよい。エンドコードは、1つの楽曲データの終了を示すデータである。すなわち、この実施例では異なる曲の楽曲データが連続して多数記憶されていることから、該エンドコードによって個々の楽曲データの終了を示すようにしている。

【0020】なお、上記楽曲データには複数チャンネルのデータが混在しており、各チャンネルはそれぞれ異なる演奏パート（例えば、左手で演奏するパート、右手で演奏するパート等）に対応する。そこで、楽譜表示を行う際には、各パート別に楽譜を表示するようにしてもよい。なお、楽曲データに定義されない音部記号（つまり、ト音記号やヘ音記号など）については、楽譜を表示するパート毎のノートイベントデータ（詳しくは音高）に従って自動的に選択した音部記号を表示する。

【0021】上述した電子楽器においては、CPU1によって楽曲データに基づいて自動演奏される楽曲の楽譜

を表示するために各種処理が行われる。すなわち、楽譜表示などの各種処理は、CPU1が所定のプログラム（ソフトウェア）を実行することにより実施される。勿論、コンピュータソフトウェアの形態に限らず、DSP（ディジタル・シグナル・プロセッサ）によって処理されるマイクロプログラムの形態でも実施可能であり、また、この種のプログラムの形態に限らず、ディスプレイ回路又は集積回路若しくは大規模集積回路あるいはゲートアレイ等を含んで構成された専用ハードウェア装置の形態で実施してもよい。

【0022】そこで、まず楽譜表示や演奏ガイドあるいは自動演奏などの各種処理を含む「メイン処理」について、図3を用いて説明する。図3は、該電子楽器におけるCPU1で実行する「メイン処理」の一実施例を示したフローチャートである。該「メイン処理」は、電子楽器の電源オンと同時に開始し、電子楽器の電源オフと同時に終了する処理である。

【0023】ステップS1では、初期設定を行う。すなわち、電子楽器への電源投入により所定のプログラムの実行を開始して、RAM3の内容をクリアするなどの各種の初期設定（つまり初期化）を行う。この実施例で行われる初期設定としては、例えばRAM3のレジスタやフラグ、テーブル、メモリあるいはパラメータ等の内容をクリアする（あるいは予め設定された初期値に戻す）、MIDIインタフェース10や通信インタフェース11やタイマ1A等をクリアする、さらにはディスプレイ7Aに所定の初期画面を表示するなどがある。ステップS2では、「パネルスイッチ処理」を行う。この「パネルスイッチ処理」は、ユーザ選択に従って外部記憶装置9などから楽曲データを読み出す処理、楽曲の楽譜を表示するか否かを選択する処理、演奏ガイドを行うか否かを選択する処理、あるいは自動演奏の開始及び終了を指示する処理、さらには自動演奏時に用いる演奏条件を入力するなどのその他の設定処理等、該電子楽器で実行する各種処理をパネル操作子5Aを用いて設定指示する処理である（詳しくは、後述する図4で説明する）。

【0024】ステップS3では、「楽譜表示処理」を行う。この「楽譜表示処理」は、ユーザにより選択された楽曲データに基づいて予め生成される頁毎の楽譜を、楽曲の進行に応じて順次にディスプレイ7Aに表示する処理である（詳しくは、後述する図8で説明する）。また演奏ガイドを行う選択がなされている場合には、楽曲の進行に応じて演奏がなされている楽譜上の現在位置を指し示すように、所定の表示態様のカーソルを移動表示する。カーソルが表示中の楽譜の最後まで達した場合には、現在表示中の頁の楽譜を次の頁の楽譜とを入れ替えて表示し、カーソルを入れ替えに表示した楽譜の先頭位置に戻す。後述するように1小節内に含まれる楽譜記号が複数の基本表示ブロックに分割されて表示された場合

であっても、このカーソルを1小節毎の表示長さに応じて移動速さを変化させながら表示することによって、該カーソルは常に自動演奏が行われている現在の正確な楽譜上の位置を指し示す。これにより、楽譜表示態様に関わらずユーザに対して正確に演奏タイミングを指示することができる。このように、この実施例では、ユーザに対する演奏補助をディスプレイ7Aに表示された楽譜上のカーソルを移動することによって行う。

【0025】ステップS4では、演奏ガイド・自動演奏処理を行う。演奏ガイド処理は、ユーザ選択された楽曲データに基づき、選択された演奏ガイドモード（例えば片手ガイド、両手ガイド、ガイドなしの各モード）に応じた演奏ガイドを行う。例えば、「片手ガイドモード」として左手ガイド（あるいは右手ガイド）が選択されている場合には、楽曲データ中の左手パート（あるいは右手パート）の楽曲データに従って演奏操作子5Aに対応して配置されている発光ダイオード（LED）4Aを点灯・消灯することによって、左手（あるいは右手）で操作する演奏操作子5Aの演奏指示をユーザに対して行うことができるようになっている。すなわち、この実施例では、ディスプレイ7Aに表示された楽譜上をカーソルが移動することによってユーザに対する演奏補助を行うようにすると共に、演奏操作子5Aに対応して配置された発光ダイオード（LED）4Aを点灯・消灯することによってもユーザに対する演奏補助を行うことができるようになっている。上記した演奏ガイド処理と共に実行される自動演奏処理は、ユーザにより選択された楽曲データに基づき、演奏ガイドモードに応じた演奏態様で自動演奏を行う。例えば、「片手ガイドモード」として左手ガイド（あるいは右手ガイド）が選択されている場合には左手パート（あるいは右手パート）の演奏のみをミュートする（つまり楽曲データの再生は行うが発音は行われない）、「両手ガイドモード」が選択されている場合には左手パート及び右手パートの両方の演奏をミュートする、「ガイドなしモード」が選択されている場合には全てのパートの演奏を行う。ステップS5では、「その他の処理」を行う。すなわち、上記した以外の各種処理を行う。これらステップS1～S5までの各処理は、当該電子楽器の電源がオフされるまで繰り返し実行される。

【0026】次に、上述した「メイン処理」において実行する「パネルスイッチ処理」（図3のステップS2参照）の詳細な動作について、図4を用いて説明する。図4は、「パネルスイッチ処理」の一実施例を示したフローチャートである。以下、図4のフローチャートに従って、当該処理における処理動作を説明する。

【0027】ステップS11では、楽譜表示や演奏ガイド及び／又は自動演奏の対象となる楽曲データの選択を行う。該楽曲データの選択においては、楽譜表示や演奏ガイド及び／又は自動演奏の対象とする楽曲データとし

て、外部記憶手段9（例えば、ハードディスクなど）に記憶されている既存の楽曲データの中からいずれかの楽曲データを選択して使用するか、あるいは新たに楽曲データをダウンロードして使用するかを指定する。新たに楽曲データをダウンロードする場合には、ダウンロード先（例えば、他のMIDI機器10Aやサーバコンピュータ11Aなど）を選択し、該ダウンロード先に記憶されている楽曲データの一覧からユーザ所望の楽曲データを選択する。楽曲データを選択すると、該楽曲データのダウンロードをスタートし、ダウンロード先から該電子楽器の外部記憶手段9（例えば、ハードディスクなど）などに楽曲データが送信されて記憶される。ステップS12では、楽譜表示モードの選択を行う。すなわち、ユーザが演奏される楽曲の進行に応じた楽譜の表示を行うか否かのどちらを選択したかの判定を行う。楽譜表示モードが選択されている場合（つまり、ユーザが楽譜表示を行うと選択した場合）には、さらに楽譜の表示サイズの選択が行われる。この楽譜の表示サイズの選択においては、例えば2小節分の楽譜を1段として4段分を同時に表示することのできる表示サイズで（つまり8小節分の楽譜が1頁分として表示される）表示するか、4小節分の楽譜を1段として4段分を同時に表示することのできる表示サイズで（つまり16小節分の楽譜が1頁分として表示される）表示するかをいずれかを選択することができる。

【0028】ステップS13では、演奏ガイドモードの選択を行う。すなわち、ユーザが演奏ガイドを行うか否かのどちらを選択したかの判定を行う。演奏ガイドモードが選択されている場合（つまりユーザが演奏ガイドを行うと選択した場合）には、さらに演奏ガイドモードの種類の選択が行われる。この演奏ガイドモードの種類には、上述したように「片手ガイドモード」、「両手ガイドモード」などがある。演奏ガイドモードの種類として「片手ガイドモード」が選択された場合には、左手あるいは右手（つまり左手パート又は右手パートのいずれか1パート）のみに対する演奏ガイドが楽曲の進行に応じて行われる。演奏ガイドモードの種類として「両手ガイドモード」が選択された場合には、左手と右手の両手（つまり左手パート及び右手パートの2パート）に対する演奏ガイドが楽曲の進行に応じて行われる。ステップS14では、自動演奏モードの選択を行う。すなわち、ユーザが自動演奏を行うか否かのどちらを選択したかの判定を行う。自動演奏モードが選択されている場合（つまりユーザが自動演奏を行うと選択した場合）には、自動演奏スタートスイッチを押下すると自動演奏が開始されるし、自動演奏ストップスイッチを押下すると自動演奏が停止される。ステップS15では、その他の設定処理が行われる。該その他の設定処理としては、楽譜表示や演奏ガイドあるいは自動演奏のためのテンポ設定あるいは自動演奏の際に用いる各種演奏条件の設定などの、

パネル操作子6Aの操作状態の検出に従って行われる各種設定処理などがある。

【0029】ステップS16では、自動演奏が行われていない場合に（つまり自動演奏停止中に）、自動演奏スタートスイッチが押下操作されたか否かの判定を行う。すなわち、自動演奏の開始が指示されたか否かを判定する。自動演奏スタートスイッチが押下された場合には（ステップS16のYES）、「表示楽譜データ作成処理」を行う（ステップS17）。該「表示楽譜データ作成処理」では、自動演奏される楽曲の進行にあわせてディスプレイ7A上に楽譜を頁単位に表示するための表示楽譜データを生成する。この「表示楽譜データ作成処理」の詳しい説明については、後述する（後述の図5参照）。ステップS18では、楽曲データに基づいた自動演奏を開始すると共に、楽譜表示モードが選択されている場合には（ステップS12参照）、楽譜表示を開始する。演奏ガイドモードが選択されている場合には（ステップS13参照）、演奏ガイドを開始する。すなわち、楽譜表示モードが選択されている場合には、生成した表示楽譜データに基づいて最初の頁の楽譜を表示する。演奏ガイドモードが選択されている場合には、楽曲データに基づいて演奏ガイドを開始する。一方、ステップS16において自動演奏スタートスイッチが押下されていない場合には（ステップS16のNO）、ステップS19の処理へ飛ぶ。すなわち、自動演奏の開始が指示されていない場合には、自動演奏や演奏ガイドは勿論行われなし、またこうした場合には楽譜表示を行う必要もないことから表示楽譜データを生成しなくてよい。そこで、上述のステップS17及びステップS18の処理を実行することなく、ステップS19の処理へ飛ぶ。ステップS19では、楽曲データの再生中（つまり自動演奏中）に自動演奏ストップスイッチが押下されたか否かを判定する。自動演奏ストップスイッチが押下されている場合には（ステップS19のYES）、現在行われている自動演奏を停止すると共に、該自動演奏と共に行われている楽譜表示や演奏ガイドなどを同時に停止する（ステップS20）。自動演奏ストップスイッチが押下されていない場合には（ステップS19のNO）、上記ステップS20の処理は行わない。すなわち、自動演奏が行われている場合には、自動演奏、演奏ガイド、楽譜表示を引き続き実行する。

【0030】次に、上述した「パネルスイッチ処理」において実行する「表示楽譜データ作成処理」（図4のステップS17参照）の詳細な動作について、図5を用いて説明する。図5は、「表示楽譜データ作成処理」の一実施例を示したフローチャートである。以下、図5のフローチャートに従って、当該処理における処理動作を説明する。

【0031】ステップS21では、楽曲データに含まれる初期設定イベントデータに記載されている（あるいは

自動検出された）拍子や調号を取得することで、1小節分の楽曲データに対応する調号及び拍子記号を検出する。また、該取得した拍子に基づいて1小節分に対応する楽曲データに含まれるノートイベントデータを取得して音符記号や休符記号等を検出する。例えば、初期設定イベントデータに記憶された拍子記号の値が4/4拍子であるとする、ノートイベントデータのゲートタイム及びデュレーションデータとに基づいて4/4拍分の長さに対応する楽曲データのノートイベントデータから音符を検出したり、該検出した音符とデュレーションデータとに基づいて休符を検出したりする。これにより、1小節分の曲構成がどのようなものであるかを得ることができる。次に、こうして得られた1小節分の曲構成から、各拍毎に含まれる楽譜記号（つまり、調号、拍子記号、音符記号、休符記号など）を全て表示するのに必要とされる表示幅（例えばドット数）を求める（ステップS22）。

【0032】ステップS23では、「1小節楽譜表示の分析処理」を行う。すなわち、楽譜表示を行う場合、通常は1つの基本表示ブロックに1小節分の楽譜記号の全てを表示するのであるが、1小節内にあまりにもたくさんの楽譜記号が入っている場合には、こうした全ての楽譜記号を該基本表示ブロック内に重ねることなく収めて表示することが困難である。このような場合、1小節分の楽曲データに含まれる楽譜記号を1つの基本表示ブロックだけに配置して表示するのではなく、複数の基本表示ブロックに分割配置して表示する。この際に考慮すべき楽譜記号は、五線譜上に表示する音部記号、拍子記号、調号、音符記号（符尾情報や符頭情報などを含む）、休符記号である。そこで、これらの各記号を表示するのに必要な表示幅を各記号毎に求め、各記号の表示幅の合計が基本表示ブロック幅に収まっているかを確認し、収まっていなければ複数の基本表示ブロックに分割して配置するように、1小節分の楽譜記号を分割する。該「1小節楽譜表示の分析処理」の具体的な処理についての詳細な説明は後述する（後述の図6参照）ことから、ここでの詳細な説明を省略する。

【0033】ステップS24では、現在分析処理中の小節番号（CMN）に1を加算する。ステップS25では、現在分析処理中の小節番号（CMN）が楽曲データ全体の総小節数（SSN）以下であるか否かの判定を行う。現在の小節番号（CMN）が総小節数（SSN）以下である場合（ステップS25のYES）、上記ステップS23の処理で行った分析結果に基づいて1小節分の表示楽譜データを作成し、RAM3の所定の記憶エリアに書き込む（ステップS26）。このRAM3への表示楽譜データの書き込みが終了すると、次の小節分の表示楽譜データを作成するためにステップS21の処理に戻る。一方、ステップS25において、現在分析処理中の小節番号（CMN）が総小節数（SSN）より大きい場

合には(ステップS25のNO)、楽曲データ全体にわたる全小節についての表示楽譜データの作成が終了していることから該「表示楽譜データ作成処理」を終了する。こうすることによって、1小節分毎に楽曲データに基づいて楽譜表示を行う際に用いられる表示楽譜データが生成されてRAM3に記憶される。この際に、所定単位毎に頁数が付与されて表示楽譜データは記憶される。また、その際に総頁数(PN)も同時に表示楽譜データに記憶される。後述する「楽譜表示処理」(後述の図8参照)では、楽曲の進行に応じてRAM3に記憶された表示楽譜データを1頁ずつ順次に読み出すことによって楽譜表示を行っている。

【0034】ここで、上述した「表示楽譜データ作成処理」において実行する「1小節楽譜表示の分析処理」

(図5のステップS23参照)の詳細な動作について、図6及び図7を用いて説明する。図6は、「1小節楽譜表示の分析処理」の一実施例を示したフローチャートである。図7は、図6に示した「1小節楽譜表示の分析処理」で使用するパラメータ等を説明するための概念図である。ただし、この図7に示した概念図は、1小節分の楽譜記号の全てが第1基本表示ブロックに収まる場合を例示的に図示したものである。以下、図6のフローチャート及び図7の概念図に従って、当該「1小節楽譜表示の分析処理」の処理動作について簡単に説明する。

【0035】まず、ステップS31では、上述した「表示楽譜データ作成処理」で求められた(図5のステップS22参照)音部記号と調号と拍子記号とを表示するために必要な表示幅及び各拍の表示に必要な幅を各々記憶しておく。図7から理解できるように、この実施例では音部記号(図7ではト音記号)と調号(図7では記載なし)と拍子記号(図7では4/4拍子)とを表示するために必要な表示幅をヘッダー表示幅Head1(つまり、Head1=音部記号の幅+調号の幅+拍子記号の幅)として、各拍内に含まれる全ての楽譜記号を表示するのに必要な幅を拍表示幅d[n](nは拍数)(図7ではd

[1]~d[4])として各々記憶する。また、この他に

$$\text{Idx}=\text{Idx1}=\text{Start}\cdots(\text{式2-1})$$

$$\text{Diff}=\infty\cdots(\text{式2-2})$$

$$\text{Head2}=0\text{または(音部記号の幅と調号の幅の和)}\cdots(\text{式2-3})$$

$$\text{Width1}=\text{Width}-\text{Head1}\cdots(\text{式2-4})$$

$$\text{Width2}=\text{Width}-\text{Head2}\cdots(\text{式2-5})$$

小節途中で該小節を分割して表示するために表示段が変わってしまう場合には、新たに音部記号や調号などを次段の先頭位置に表示しなければならない。そこで、Head2(式2-3参照)を予め算出しておく。上記式2-3で示すように、Head2に対しては0または(音部記号の幅と調号の幅の和)のいずれかがセットされる。すなわち、分配を行った際に、同じ段の基本表示ブロックに分配が行われる場合には音部記号や調号を挿入して表示する必要がないので該Head2に0をセットする。一方、異

も、1小節を表示するのに最大限確保することのできる表示幅を基本表示ブロック幅Widthとして記憶する。この他にも、表示先頭拍Startとして「1」拍を、表示終了拍Endとして1小節中における最終拍(図7では「4」拍)をそれぞれセットしておく。ステップS32では、ヘッダー表示幅Head1と表示先頭拍Startから表示終了拍Endまでの全ての拍表示幅d[1]~d[n]とを合計した結果と基本表示ブロック幅Widthとを比較して、ヘッダー表示幅Head1と表示先頭拍Startから表示終了拍Endまでの全ての拍表示幅d[1]~d[n]とを合計した結果が基本表示ブロック幅Width以下であるか否かを判定する。又は、表示先頭拍Startと表示終了拍Endとが等しいか否かを判定する。

【0036】ヘッダー表示幅Head1と表示先頭拍Startから表示終了拍Endまでの全ての拍表示幅d[1]~d[n]とを合計した結果が基本表示ブロック幅Width以下であると判定された場合には(ステップS32のYES)、該基本表示ブロックに表示先頭拍Startから表示終了拍Endまでの全ての楽譜記号を表示することができると判定されることから、表示先頭拍Startから表示終了拍Endまでの全ての楽譜記号を該基本表示ブロックに入れる(ステップS33)。又は、表示先頭拍Startと表示終了拍Endとが等しいと判定された場合(ステップS32のYES)、この場合には全ての拍の分析処理が終了したものとして、残りの最後の拍(すなわち、表示先頭拍Startと表示終了拍Endとが等しくなるため)に含まれる楽譜のみを該基本表示ブロックに入れる(ステップS33)。

【0037】他方、ステップS32において、ヘッダー表示幅Head1と表示先頭拍Startから表示終了拍Endまでの全ての拍表示幅d[1]~d[n]とを合計した結果が基本表示ブロック幅Width以下であると判定された場合、又は、表示先頭拍Startと表示終了拍Endとが等しくない判定された場合(ステップS32のNO)、以下に示した(式2-1)~(式2-5)の各式に従って各値を設定する(ステップS34)。すなわち、

なる段の基本表示ブロックに分配が行われる場合には次段の基本表示ブロックの初めには音部記号や調号などを自動的に挿入する必要があることから、該Head2に音部記号の幅と調号の幅の和をセットする。このように予めセットしておくことで、分配が多段にわたる場合においても、各段の先頭位置に音部記号や調号などを自動的に表示することができ、楽譜表記のルールに従った楽譜表示を行うことができるようになっている。あるいは、楽曲の途中で拍子などが変わる場合(この場合には楽曲デ

ータに該当する拍子イベントデータが記憶されている)にはそうした調号や拍子記号などを楽譜に表記しなければならぬことから、こうした場合においてもHead 2に音部記号の幅と調号の幅の和をセットしておく。ただし、カデンツァ(すなわち、主として曲のエンディング部分に使われる無伴奏のソロ)と判断される場合には、楽曲データに拍子イベントデータが記憶されていたとしても、楽譜表記のルールに従うように調号や拍子記号などを楽譜に表記しなくてよいことから、こうした場合にはHead 2に0をセットする。このカデンツァの判断は、例えば4/4拍子の曲である場合には9/4拍子以上の拍子イベントデータが現れた場合、8/8拍子の曲である場合には17/8拍子以上の拍子イベントデータが現れた場合などにカデンツァと判断する。

【0038】さらに、表示先頭拍Startから拍Idxまでの拍を表示するのに必要な表示幅(これを幅D1とする)及び拍Idxの次の拍(つまりIdx+1)から表示終了拍Endまでの拍を表示するのに必要な表示幅(これを幅D2とする)を求める(ステップS35)。すなわち、表示先頭拍Startから表示終了拍Endまでを表示するのに必要な表示幅を拍Idxを区切りとして分割幅D1と分割幅D2とに分割する。ステップS36では、算出した分割幅D1と表示残幅Width1とを比較する。分割幅D1が表示残幅Width1よりも大きい場合には(ステップS36のYES)、表示先頭拍Startから拍Idx1までの拍が今の基本表示ブロックに入るように設定される(ステップS37)。それと共に、表示開始拍Startに(Idx+1)を、ヘッダー表示幅Head1にHead2をそれぞれ再設定する。このステップS37の処理を終了すると、ステップS32の処理へ戻る。例えば、表示先頭拍Startから拍Idx1までの拍が第1の基本表示ブロックに入るように設定された場合には、この第1の基本表示ブロックに続く第2の基本表示ブロックを新たに第1の基本表示ブロックとして、上記処理を続ける。

【0039】上述したステップS36において、分割幅D1が表示残幅Width1よりも大きくない場合には(ステップS36のNO)、分割幅D2と表示残幅Width2との比較を行う(ステップS38)。分割幅D2が表示残幅Width2よりも大きくない場合には(ステップS38のNO)、分割幅D1と分割幅D2との幅の差が変数Diff以上であるか否かが判定される(ステップS39)。分割幅D1と分割幅D2との幅の差が変数Diff以上である場合には(ステップS39のYES)、上記したステップS37の処理へ行き、表示先頭拍Startから拍Idx1までの拍が今の表示ブロックに入るように設定されると共に、表示開始拍Startに(Idx+1)を、ヘッダー表示幅Head1にHead2をそれぞれ再設定する。そして、ステップS32の処理へ戻る。分割幅D1と分割幅D2との幅の差が変数Diff以上でない場合には(ステップS39のNO)、変数Diffに分割幅D1と分割幅D2

との幅の差(すなわち、差異)を設定する(ステップS41)。該ステップS41の処理終了後、あるいは分割幅D2が表示残幅Width2よりも大きい場合には(ステップS38のYES)、新たに拍Idx1に対して拍Idxの値を、拍Idxに対して「Idx+1」を、それぞれ再設定(ステップS40)してからステップS35の処理へ戻る。

【0040】ここで、上述したステップS32からステップS41までの各処理に関する基本的な考え方を説明する。小節を構成する複数の拍に対応する楽譜表示を、拍単位で複数の基本表示ブロックに分配するときには、隣り合う2つの基本表示ブロックの楽譜表示がなるべく均等な配置密度となるように、複数通りの分配組み合わせ毎に分割幅D1と分割幅D2の幅の差(つまり、 $|D1 - D2|$)を求める。そして、この求めた幅の差が最も小さくなる組み合わせを選択し、選択された組み合わせに基づく分割幅D1に対応する(単数又は複数の)拍の楽譜表示を1つ目の基本表示ブロックに行うように決定する。例えば、4/4拍子のときには、分割幅D1に対応する拍と分割幅D2に対応する拍との組み合わせは、以下に示すような3つの組み合わせとなる。すなわち、第1の組み合わせは、分割幅D1に1拍目(d[1])、分割幅D2に2~4拍目まで(d[2]+d[3]+d[4])である。第2の組み合わせは、分割幅D1に1拍目と2拍目(d[1]+d[2])、分割幅D2に3拍目と4拍目まで(d[3]+d[4])である。第3の組み合わせは、分割幅D1に1~3拍目(d[1]+d[2]+d[3])、分割幅D2に4拍目(d[4])である。図6に示した「1小節楽譜表示の分析処理」におけるステップS35、S36、S38、S39、S41、S40、S35の各処理またはステップS35、S36、S38、S40、S35の各処理を繰り返して処理することによって、各組み合わせ毎に分割幅D1と分割幅D2との差を求め、最少の差異となる組み合わせを選択する。

【0041】例えば、上記第1の組み合わせにおける分割幅D1がWidth1よりも大きくない場合(ステップS36のNO)であり、かつ、分割幅D2がWidth2よりも大きい場合(ステップS38のYES)に、ステップS35が実行されると、分割幅D1には1拍目と2拍目の楽譜記号の合計幅(つまり、拍表示幅d[1]+d[2])が、分割幅D2には3拍目と4拍目の楽譜記号の合計幅(つまり、拍表示幅d[3]+d[4])がそれぞれ設定される。ステップS36にて「YES」である場合(つまり、分割幅D1がWidth1より大きい場合)には、上記第2の組み合わせにおける分割幅D1が基本表示ブロックに入らないことになり、上記第1の組み合わせにおける分割幅D1(この場合、Start=Idx1=1)が選択されるべきものとなり、ステップS37に進み、Start~Idx1の拍(この場合、1拍目(d[1])のみ)を1つ目の基本表示ブロックに入れることを決定する。

その後、ステップS32に戻って、次の基本表示ブロック、次の拍（この場合、 $d[2]$ ）から再び同様のステップに基づいて処理を始める。他方、ステップS36にて「NO」である場合（つまり、分割幅D1がWidth1より小さいか等しい場合）には、ステップS38に進む。ステップS38にて「NO」である場合（つまり、分割幅D2がWidth2より小さいか等しい場合）には、ステップS39に進む。ステップS39にて「YES」である場合（つまり、分割幅D1と分割幅D2との差が差異Diffの方が小さい場合）には、前回（つまり、 $Start=1$ 、 $dx1=1$ ）の差異Diffが一番小さいということになり、第1拍（ $d[1]$ ）を1つ目の基本表示ブロックに入れることを決定し、ステップS32に戻って、次の基本表示ブロック、次の拍（ $d[2]$ ）から再び同様のステップを始める。

【0042】一方、ステップS39にて「NO」である場合（つまり、今回の差異 $|D1-D2|$ の方が前回の差異Diffよりも小さい場合）には、ステップS41に進み、今回の差異を最小の差異Diffとする。ステップS41の処理の後、またはステップS38にて「YES」である場合（つまり、分割幅D2がWidth2より大きい場合）には、再度ステップS40を実行して、上記第3の組み合わせにおける差異 $|D1-D2|$ を求めるために、再度ステップS35に進み、同様にして処理する。このようにして、ステップS33を除いてステップS32～ステップS41までの各処理を繰り返し実行することにより、1小節を構成する複数拍の楽譜表示を、拍単位で、基本表示ブロックに入れていく。そして、ステップS32において、残りの全拍の楽譜表示が1つの基本表示ブロックに入るか、または最後の拍に達した場合には、残りの全拍または最後の拍の楽譜表示を現在の基本表示ブロックに入れることを決定して、1小節分の分析処理を終了する。

【0043】以上のようにすることによって、1小節内に含まれる全ての楽譜記号が1つの基本表示ブロック内に入りきらないような場合には、1小節内に含まれる楽譜記号を複数の基本表示ブロックに表示するように拍毎に楽譜記号を分割して複数の基本表示ブロックに配置する。また、単に楽譜記号を拍毎に分割して複数の基本表示ブロックに配置するだけでなく、各基本表示ブロックにおける楽譜記号の表示密度を調整しながら表示することが行われる。

【0044】ここで、上述した「1小節楽譜表示の分析処理」について図7を用いて具体的に説明する。例えば、基本の音符記号（例えば、全音符、2分音符、4分音符）や休符記号を表示するために必要なドット数を1記号あたり4ドット、基本の音符記号に付加する旗（例えば、8分音符や16分音符などで用いる旗表示）を表示するために必要なドット数を3ドット、基本の音符記号に付加する付点（例えば、付点4分音符や付点8分音

符などで用いる付点表示）を表示するために必要なドット数を2ドット、臨時記号（例えば調号以外につけられる \sharp や \flat などの、楽曲の途中で特定の音の高さを一時的に半音上または下に変化させる記号）を表示するために必要なドット数を5ドットとする。こうした場合に、図7に示した楽譜の各拍を表示するのに必要な表示幅を求めると、第1拍目（つまり $d[1]$ ）は21ドット（つまり、 $5（臨時記号） \times 2 + 4（音符記号） + 3（旗：付点の2ドットは旗の表示内に含まれる）$ ）、第2拍目（つまり $d[2]$ ）及び第3拍目（つまり $d[3]$ ）は共に16ドット（つまり、 $4（音符記号） \times 4$ ）、第4拍目（つまり $d[4]$ ）は11ドット（つまり、 $4（音符記号） + 3（旗） + 4（休符記号）$ ）となる。したがって、図7に示したような第1小節分の楽譜を表示するのに必要な表示幅は全部で64ドット（つまり、 $21+16+16+11$ ）である。上述したように、こうして算出された1小節分を表示するのに必要な表示幅が、第1基本表示ブロック幅Widthからヘッダー表示部Head1を減算した表示残幅Width1よりも小さければ、第1基本表示ブロック内に全ての楽譜記号を表示することができる。例えば、表示残幅Width1として80ドットあるような場合には、第1小節を表示するのに必要な表示幅64ドットが全て第1基本表示ブロック内に収まることから、こうした場合には楽譜記号を分割して表示する必要がなく全ての楽譜記号を第1基本表示ブロックに表示することができる。

【0045】他方、算出された1小節分を表示するのに必要な表示幅が、第1基本表示ブロック幅Widthからヘッダー表示部Head1を弾いた表示残幅Width1よりも大きければ、第1基本表示ブロック内に全ての楽譜記号を表示することができないことから、こうした場合には1小節内の楽譜記号を分割して、複数の基本表示ブロックWidthに分配する必要がある。例えば、表示残幅Width1として50ドットしかない場合には、第1小節を表示するのに必要な表示幅64ドットが全て収まらないために、この場合には楽譜記号を分割して表示する必要がある。こうした場合における分割としては、1拍及び2拍目の37ドット（つまり、 $21+16$ ）を第1基本表示ブロックに分配し、3拍及び4拍目の27（つまり、 $16+11$ ）ドットを第2基本表示ブロックに分配する。あるいは第1拍目の21ドットを第1基本表示ブロックに、第2～第4拍目までの43ドット（つまり、 $16+16+11$ ）を第2基本表示ブロックに分配してもよいが、こうした場合には第1基本表示ブロックと第2基本表示ブロックに表示される楽譜記号が偏って表示されることになり、ユーザにとって非常に見にくくなることから、こうした分配は行われない。なお、1拍～3拍目までの52ドット（つまり、 $21+16+16$ ）を第1基本表示ブロックに分配し、4拍目の11ドットのみを第2基本表示ブロックに分配することは、1拍～3拍目までが表示残幅Width1の50ドットを超えてしまうことから、こうした分配はでき

ない。

【0046】表示残幅width1が50ドット、各拍の楽譜記号を表示するのに必要なドット数がそれぞれ60、15、15、15ドットであるような場合にも、第1小節内の全ての楽譜記号を表示するのに必要な表示幅105ドット（つまり、60+15+15+15）が第1基本表示ブロック内に全て収まらないために分割して表示する必要がある。しかし、この場合に1拍目の60ドットを第1基本表示ブロックに分配し、2拍～4拍目までの45ドット（つまり、15+15+15）を第2基本表示ブロックに分配することは、1拍目のみで表示残幅width1の50ドットを超えてしまうことから、こうした分配はできない。かといって、1拍及び2拍目（つまり、60+15の75ドット）と3拍及び4拍目（つまり、15+15の30ドット）、あるいは1拍～3拍目まで（つまり、60+15+15の90ドット）と4拍目のみ（15ドット）のように分割しても初めの分割側のみで表示残幅width1の50ドットを超えてしまうことから、こうした分配もできない。このように1拍でも第1基本表示ブロックに入りきらないことが生じたような場合には、1拍分をそれ以上分割することができないことから、1拍目の全ての楽譜記号を第1基本表示ブロックに割り当てて表示するようにする。こうした場合、1拍目の楽譜記号を表示するのに必要な表示幅は60ドットであり、これは第1基本表示ブロックの表示残幅width1の50ドットを超えてしまうことから、一部分の楽譜記号は重ねて表示されることになる。

【0047】また、第1基本表示ブロック及び第2基本表示ブロックの表示残幅が共に50ドットで、5/4拍子における各拍の楽譜記号を表示するのに必要なドット数がそれぞれ20、25、40、20、10ドットであるような場合にも、第1小節を表示するのに必要な表示幅115ドット（つまり、20+25+40+20+10）が全て第1基本表示ブロックに収まらないために分割して表示する必要がある。しかし、この場合に1拍目の20ドットを第1基本表示ブロックに分配し、2拍～5拍目の95ドット（つまり、25+40+20+10）を第2基本表示ブロックに分配することは、2拍～5拍目が第2基本表示ブロックの表示残幅width2の50ドットを超えてしまうことから、こうした分配はできない。1拍及び2拍目の45ドット（つまり、20+25）を第1基本表示ブロックに分配し、3拍～5拍目の70ドット（つまり、40+20+10）を第2基本表示ブロックに分配することは、3拍～5拍目が第2基本表示ブロックの表示残幅width2の50ドットを超えてしまうことから、こうした分配もできない。さらに1拍～3拍目の85ドット（つまり、20+25+40）を第1基本表示ブロックに分配し、4拍及び5拍目の30ドット（つまり、20+10）を第2基本表示ブロックに分配することは、1拍～3拍目が第1基本表示ブロックの表示残幅width1の50ドットを超えて

しまうことから、こうした分配もできない。したがって、こうした場合には、第1基本表示ブロックに最大限入りきる拍までを第1基本表示ブロックに、残りの拍をさらに分割してその他の基本表示ブロックに分配する。すなわち、この例では、1拍及び2拍目（つまり、20+25の45ドット）を第1基本表示ブロックに割り当てる。残りの3拍～5拍目（つまり、40+20+10の70ドット）をさらに分割して処理すると、結果として1拍及び2拍目（つまり、20+25の45ドット）が第1基本表示ブロックに、3拍目（40ドット）が第2基本表示ブロックに、4拍及び5拍目（つまり、20+10の30ドット）が第3基本表示ブロックにそれぞれ割り当てられることになる。

【0048】次に、上述した「メイン処理」において実行する「楽譜表示処理」（図2のステップS4参照）の詳細な動作について、図8を用いて説明する。図8は、「楽譜表示処理」の一実施例を示したフローチャートである。以下、図8のフローチャートに従って、当該処理における処理動作を説明する。

【0049】まず、ステップS51では、ユーザ選択された楽曲データを現在再生中であるか否か、つまり自動演奏スタートスイッチが押下されて選択された楽曲データに基づく自動演奏が行われている最中であるか否かの判定を行う。楽曲データを現在再生中でない場合には（ステップS51のNO）、楽曲の進行にあわせて楽譜表示する必要がないことから以下に示すステップS52以降の処理を行わずに、該楽譜表示処理を終了する。楽曲データを現在再生中である場合には（ステップS51のYES）、楽譜表示を行うようユーザ選択が行われているか否か（図3のステップS12参照）を判定する（ステップS52）。楽譜表示を行うよう選択されていない場合には（ステップS52のNO）、楽譜表示する必要がないことから以下に示すステップS52以降の処理を行わずに、該楽譜表示処理を終了する。楽譜表示を行うよう選択されている場合には（ステップS52のYES）、所定時間が経過したか否かを判定する（ステップS53）。所定時間が経過していない場合には（ステップS53のNO）、現在の楽譜表示を変更するタイミングでないと判断して該楽譜表示処理を終了する。すなわち、この場合にはカーソルを含めて現在表示されている状態の楽譜がそのまま表示され続ける。所定時間が経過している場合には（ステップS53のYES）、表示されている楽譜上のカーソル表示位置を所定時間分だけ移動して、現在自動演奏中の楽譜における現在位置を示すようにする（ステップS54）。

【0050】ステップS55では、カーソルが表示楽譜の最後まで達したか否かを判定する。カーソルが表示楽譜の最後まで達していない場合には（ステップS55のNO）、該表示楽譜を次の楽譜に変更する必要がないことから、該楽譜表示処理を終了する。カーソルが表示楽

譜の最後まで達した場合には（ステップS55のYES）、該楽曲データが終了していないか否かを判定する（ステップS56）。楽曲データが終了している場合には（ステップS56のNO）、自動演奏の終了にあわせて楽譜表示を終了する（ステップS61）。楽曲データが終了していない場合には（ステップS56のYES）、現在楽譜表示中のページ数に「1」を加算する（ステップS57）。すなわち、次の楽譜ページに進む。ステップS57では、現在楽譜表示中のページ数が総ページ以下でない場合には（ステップS57のNO）、自動演奏の終了にあわせて楽譜表示を終了する（ステップS59）。現在楽譜表示中のページ数が総ページ以下である場合には（ステップS57のYES）、次のページの楽譜を表示して、それと同時にカーソルを該楽譜の先頭位置に戻して表示する（ステップS60）。こうすることによって、上述した「表示楽譜データ作成処理」（図5参照）で作成されるページ毎の楽譜表示に従って楽譜表示が行われる。

【0051】なお、拍毎に分割を行う際において、連符に関してはなるべく分割を行わないで表示するほうがユーザの見やすさの点から望ましい。なお、上述の実施例においては、自動演奏開始指示に従って曲全体の楽譜を頁単位に作成するようにしたがこれに限らず、自動演奏処理を行う前に楽曲データを所定区間ずつ先読みし、該先読みした楽曲データに従って曲の所定区間ずつの楽譜を生成するようにしてもよい。また、自動演奏開始指示に従って楽譜を生成するようにしたがこれに限らず、パネル操作子として楽譜表示に使用する頁単位の表示楽譜データを生成するための専用のスイッチを設けておき、該スイッチの操作に応じて予め表示楽譜データのみを生成するようにしてもよい。なお、上述の実施例においては、1つの基本表示ブロックに1小節分の楽譜記号の全てを表示するものについて説明したがこれに限らない。例えば、1つの基本表示ブロックに1/2小節分の楽譜記号（すなわち、1小節分の楽譜記号は2つの基本表示ブロックを使用して表示される）を表示するといったように、複数の基本表示ブロックを用いて1小節分の楽譜記号を表示するようにしてもよい。こうすると、例えば1小節分の楽譜記号が多いときには3又は4つの基本表示ブロックを用いて表示を行うようにすればよいし、1小節分の楽譜記号が少ないときには1つの基本表示ブロックを用いて表示を行う、といったように適宜に1小節内に含まれる楽譜記号の全てを表示するのに必要な表示幅を可変することが簡単にできるようになることから、非常に便利である。

【0052】なお、上述の実施例では片手パートのみを楽譜表示した場合についてのみ説明したがこれに限らず、両手パートを楽譜表示する際にも上述した処理により楽譜表示するようにしてよい。また、こうした場合において、各拍を表示する際に必要な表示幅は大きな表示

幅を必要とするものに合わせる。すなわち、大譜表を表示する場合には、上下で同じタイミングに音符がある場合には、同じ横位置に音符を配置しなければならないので、上下で拍毎の表示幅を統一する必要がある。そこで、異なる演奏パート間で同じ演奏タイミングの音符が複数ある場合には、最も大きい表示幅を必要とする音符をその時間の代表とする。例えば、右手パートが60ドット必要であり、左手パートが25ドット必要であるとした場合、上述の「1小節楽譜表示の分析処理」において用いる表示幅は右手パートの60ドットになる。こうすることによって、左手パートと右手パートを演奏する際の演奏タイミングを異ならせることなく楽譜表示することができ。

【0053】なお、本実施例に係る楽譜表示制御装置を電子楽器に適用した場合、電子楽器は鍵盤楽器の形態に限らず、弦楽器や管楽器、あるいは打楽器等のようなタイプの形態でもよい。また、そのような場合に、音源装置、自動演奏装置等を1つの電子楽器本体に内蔵したものに限らず、それぞれが別々に構成され、MIDIインタフェースや各種ネットワーク等の通信手段を用いて各装置を接続するように構成されたものにも同様に適用できることはいうまでもない。また、パソコンとアプリケーションソフトウェアという構成であってもよく、この場合処理プログラムを磁気ディスク、光ディスクあるいは半導体メモリ等の記憶メディアから供給したり、ネットワークを介して供給するものであってもよい。さらに、カラオケ装置やゲーム装置、あるいは携帯電話等の携帯型通信端末、自動演奏ピアノ等に適用してもよい。

【0054】なお、楽曲データのフォーマットは上述した図2に示したようなデータ形式のものに限らず、他のどのようなデータ形式のものであってもよい。すなわち、イベントの発生時刻を曲や小節内における絶対時間で表した『イベント+絶対時間』形式のもの、イベントの発生時刻を1つ前のイベントからの時間で表した『イベント+相対時間』形式のもの、音符の音高と符長あるいは休符と休符長で楽曲データを表した『音高（休符）+符長』形式のもの、演奏の最小分解能毎にメモリの領域を確保し、演奏イベントの発生する時刻に対応するメモリ領域にイベントを記憶した『ベタ方式』形式のものなど、どのような形式のものであってもよい。また、複数チャンネル分の楽曲データが存在する場合は、複数のチャンネルのデータが混在した形式であってもよいし、各チャンネルのデータがトラック毎に別れているような形式であってもよい。さらに、楽曲データの処理方法は、設定されたテンポに応じて処理周期を変更する方法、処理周期は一定で自動演奏中のタイミングデータの値を設定されたテンポに応じて変更する方法、処理周期は一定で1回の処理において楽曲データ中のタイミングデータの計数の仕方をテンポに応じて変更する方法等、どのようなものであってもよい。また、メモリ上におい

て、時系列の楽曲データが連続する領域に記憶されていてもよいし、飛び飛びの領域に散在して記憶されている楽曲データを、連続するデータとして別途管理するようにしてもよい。すなわち、時系列的に連続する楽曲データとして管理することができればよく、メモリ上で連続して記憶されているか否かは問題ではない。

【0055】

【発明の効果】本発明によれば、分割して楽譜表示するようにし、また隣り合う基本表示ブロックに表示する楽譜記号の表示を均等な密度となるようにしたことから、たとえ1小節中に著しくたくさんの楽譜記号がある楽曲であっても、ユーザは確実に楽譜を視認することができるようになる、という効果が得られる。

【図面の簡単な説明】

【図1】 この発明に係る楽譜表示制御装置を内蔵した電子楽器の実施の形態を示すハード構成ブロック図である。

【図2】 図1に示した電子楽器で用いられる楽曲データのデータ構造の一実施例を示した概略図である。

【図3】 図1に示した電子楽器におけるCPUで実行する「メイン処理」の一実施例を示したフローチャートである。

【図4】 図3に示した「メイン処理」において実行する「パネルスイッチ処理」の一実施例を示したフローチャートである。

ャートである。

【図5】 図4に示した上述した「パネルスイッチ処理」において実行する「表示楽譜データ作成処理」の一実施例を示したフローチャートである。

【図6】 図5に示した上述した「表示楽譜データ作成処理」において実行する「1小節楽譜表示の分析処理」の一実施例を示したフローチャートである。

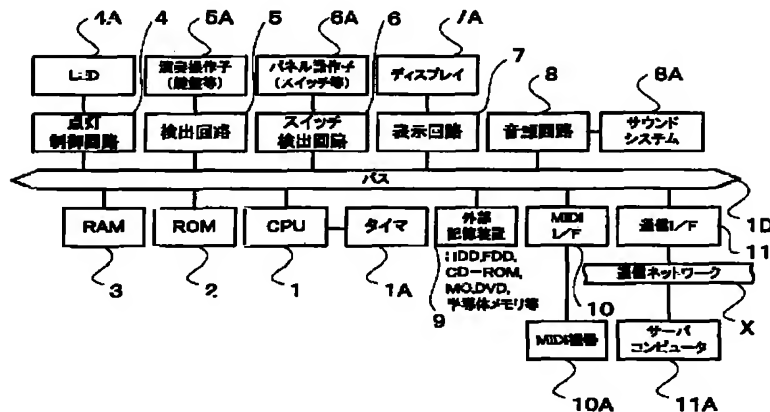
【図7】 図6に示した「1小節楽譜表示の分析処理」で使用するパラメータ等を説明するための概念図である。

【図8】 図3に示した「メイン処理」において実行する「楽譜表示処理」の一実施例を示したフローチャートである。

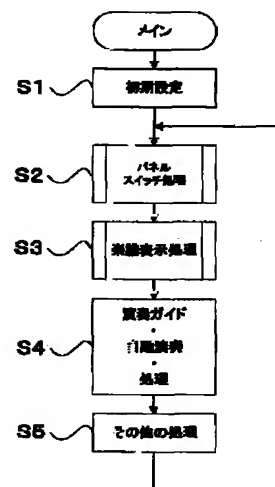
【符号の説明】

1…CPU、1A…タイマ、2…ROM、3…RAM、4…点灯制御回路、4A…LED、5…検出回路、5A…演奏操作子（鍵盤等）、6…スイッチ検出回路、6A…パネル操作子（スイッチ等）、7…表示回路、7A…ディスプレイ、8…音源回路、8A…サウンドシステム、9…外部記憶装置、10…MIDIインタフェース、10A…他のMIDI機器、11…通信インタフェース、11A…サーバコンピュータ、X…通信ネットワーク、1D…データ及びアドレスバス

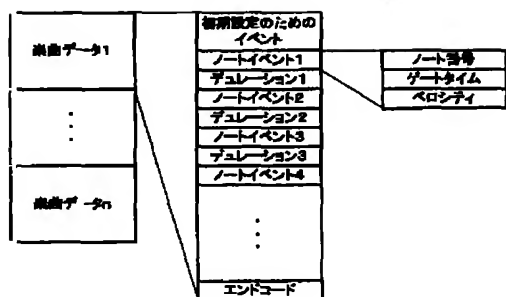
【図1】



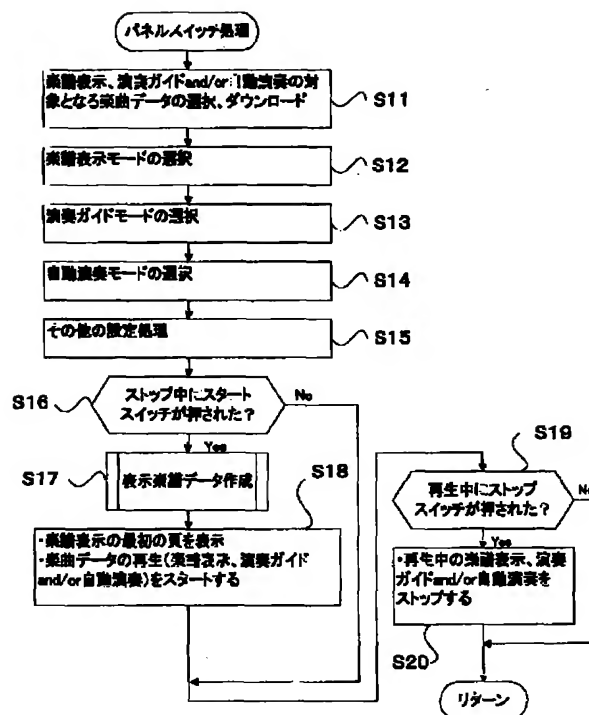
【図3】



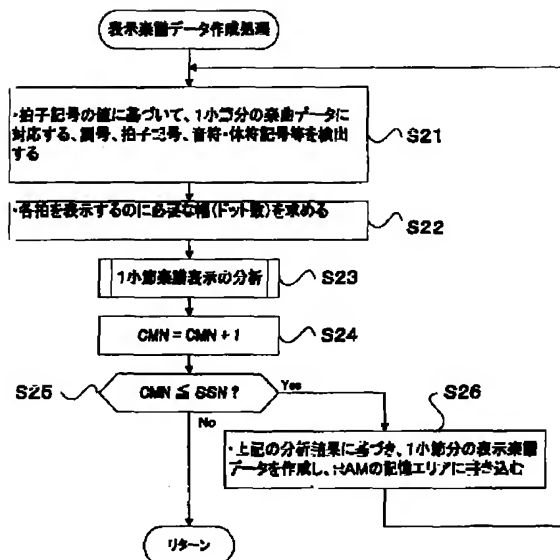
【図2】



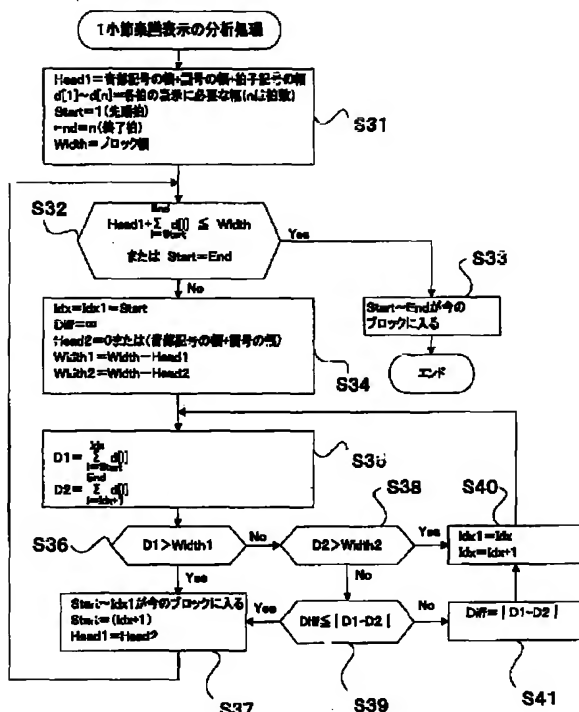
【図4】



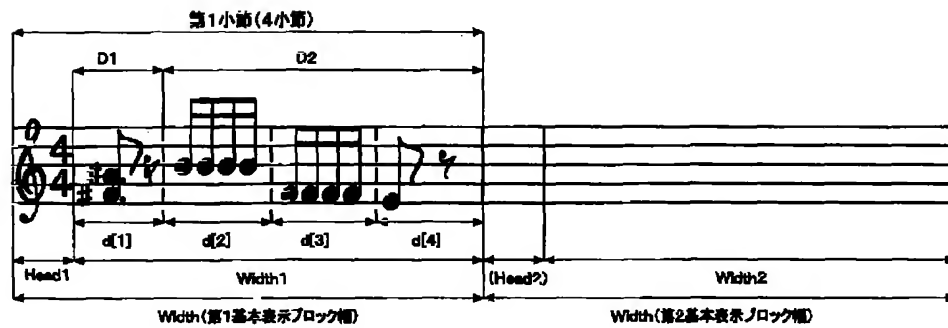
【图5】



【图6】



【図7】



【図8】

